

BUILDING AI-NATIVE BOOK DISCOVERY: FROM RETRIEVAL- AUGMENTED GENERATION TO TEXT-TO- AUDIO PRODUCTION IN A SCALABLE MICROSERVICES FRAMEWORK

¹Mrs.A. NANDINI SREE, ²NADIPI SRUJAN REDDY, ³DUSASRUHITH KUMAR, ⁴NIMMANAGOTI
KARTHIK

¹Assistant Professor, ^{2,3,4}Students, Department of Computer Science and Design, Teegala Krishna Reddy
Engineering College, Medbowli, Meerpet, Balapur, Hyderabad-500097

ABSTRACT

The ReBook – Marketplace for Used Books is a web-based application developed to provide a structured and efficient platform for exchanging used books through donation and lending. Many students and readers purchase academic and non-academic books for short-term use and later face difficulties in reselling, donating, or lending them to others. At the same time, economically disadvantaged students struggle to access affordable learning resources. Existing platforms such as social media groups and local resale markets lack proper organization, transparency, and tracking mechanisms, which often leads to confusion and unreliable communication. To overcome these limitations, the proposed system introduces a centralized marketplace where users can register, upload books, search for books using keywords, author names, ISBN, or publisher details, and send donation or lending requests. The application is developed using React for the frontend interface, Spring Boot and Spring JPA for backend processing, and MySQL for secure data storage and management. The system supports functionalities such as user authentication, book management, request approval or rejection, lending duration tracking, return confirmation, and integrated

messaging between users. The platform ensures better accountability, transparency, and ease of access through structured workflows and secure communication. By encouraging reuse of books, the system promotes environmental sustainability by reducing paper waste while also helping students minimize educational expenses. The ReBook application demonstrates the practical implementation of modern full-stack web technologies in building a scalable, reliable, and user-friendly online marketplace for used book exchange and resource sharing.

Keywords: ReBook, Used Book Marketplace, Book Donation, Lending System, React, Spring Boot, Spring JPA, MySQL, Web Application, Sustainability.

I. INTRODUCTION

In the modern digital era, online platforms have transformed the way people communicate, learn, and exchange goods and services. Educational resources such as textbooks, reference books, and competitive examination materials are essential for students and readers, yet many users face difficulties in accessing affordable books. After completing academic courses or personal reading, books often remain unused and occupy storage space instead of

being reused by others who may need them. Traditional methods such as campus resale events, second-hand bookstores, classified advertisements, and social media groups are commonly used for book exchange, but these systems lack proper organization, accountability, and transparency [1]. Existing platforms mainly focus on commercial selling rather than donation or temporary lending [2]. Offline systems also impose geographical and time limitations, making book access difficult for students from remote areas [3]. Social media platforms provide only informal communication without proper tracking mechanisms [4]. Research studies have shown that digital marketplace systems improve accessibility and simplify resource sharing [5]. Modern web applications using component-based architectures offer better user interaction and responsiveness [6]. React-based frontend systems provide dynamic rendering and reusable components for improved user experience [7]. Backend frameworks such as Spring Boot simplify API development and business logic management [8]. Spring JPA improves database interaction through object-relational mapping techniques [9]. MySQL databases provide reliable storage and efficient transaction handling for large-scale applications [10]. REST architecture enables smooth communication between frontend and backend modules [11]. Secure authentication systems are essential for protecting user data and preventing unauthorized access [12]. Structured request management workflows help maintain transparency and accountability between users [13]. Integrated messaging modules improve communication and coordination during transactions [14]. Lending duration tracking ensures proper monitoring of borrowed books and return confirmation [15].

The ReBook – Marketplace for Used Books system is proposed as a centralized web-based platform that supports both donation and lending functionalities in

a secure and organized manner. The application allows users to register, log in, upload books, manage book information, and search for books using keywords such as title, author, publisher, or ISBN [16]. Users can send requests for donation or temporary lending, while book owners can approve or reject requests based on availability [17]. The system also includes lending duration tracking and return confirmation mechanisms to maintain accountability [18]. React is used to build a responsive frontend interface with smooth navigation capabilities [19]. Spring Boot and Spring JPA manage backend processing, authentication, and REST API communication [20]. MySQL is used for storing user details, book records, request statuses, and message history securely [21]. The modular architecture improves maintainability and scalability of the application [22]. Integrated messaging functionality enables direct communication between users regarding book requests [23]. The platform promotes sustainable reuse of books, reducing paper waste and encouraging resource conservation [24]. Studies indicate that reuse-based systems contribute significantly toward environmental sustainability and affordable education [25]. The proposed system also minimizes dependency on expensive commercial book-selling platforms [26]. Dynamic search functionalities improve accessibility and reduce the time required to locate books [27]. Secure workflows enhance user trust and improve transaction reliability [28]. The application demonstrates practical implementation of full-stack development technologies in solving real-world educational resource-sharing challenges [29]. Therefore, the ReBook system provides a scalable, transparent, and user-friendly solution for exchanging used books efficiently and sustainably [30].

II. LITERATURE SURVEY

The development of online marketplaces and resource-sharing platforms has significantly improved the accessibility of educational and commercial resources. Traditional book exchange methods primarily depended on offline markets, book fairs, classified advertisements, and campus resale activities [1]. These systems required physical presence and manual communication, which limited accessibility and efficiency [2]. Earlier studies on second-hand book-selling systems mainly focused on commercial transactions rather than free donation or temporary lending [3]. Many existing platforms integrated payment gateways and commission-based models, making them unsuitable for students seeking low-cost or free educational resources [4]. Research on web-based marketplace systems demonstrated that digital platforms provide improved accessibility, faster communication, and better resource utilization [5]. React-based frontend frameworks introduced component-based architecture and virtual DOM rendering, which enhanced user interface responsiveness and performance [6]. Studies revealed that reusable UI components improve maintainability and reduce development complexity [7]. Backend development frameworks such as Spring Boot simplified REST API implementation and reduced configuration overhead [8]. Research on microservice-based backend systems showed improved scalability and modularity [9]. Spring JPA enabled efficient database interaction using object-relational mapping techniques [10]. Studies highlighted that ORM frameworks reduce manual SQL queries and improve maintainability [11]. MySQL databases were identified as reliable relational database systems for structured data management and transaction consistency [12]. RESTful communication models improved interaction between frontend and backend systems through

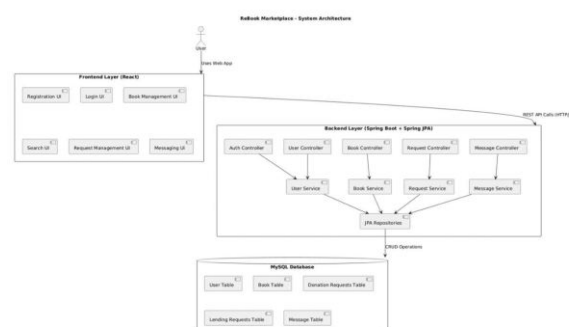
lightweight HTTP methods [13]. Software engineering principles emphasized modular architecture for improving scalability and system maintenance [14]. Authentication mechanisms were studied extensively to ensure secure access control and protection of user information [15].

Recent research focused on integrating advanced functionalities such as search optimization, request tracking, messaging, and lending workflows into marketplace applications [16]. Search algorithms based on keywords, ISBN, title, and author names improved retrieval efficiency in digital book systems [17]. Request management workflows enhanced accountability by allowing structured approval and rejection processes [18]. Studies on lending systems highlighted the importance of return tracking and status confirmation to avoid disputes between users [19]. Messaging systems integrated within applications improved coordination and reduced dependency on external communication platforms [20]. Researchers emphasized that centralized systems provide better transparency compared to social media-based communication methods [21]. Modern full-stack web development approaches using React, Spring Boot, and MySQL demonstrated high scalability and reliable performance in marketplace applications [22]. Secure authentication and validation techniques improved protection against unauthorized access and invalid data submission [23]. Research also indicated that sustainable reuse systems help reduce environmental impact by encouraging recycling and minimizing paper waste [24]. Studies on educational resource-sharing platforms concluded that low-cost book exchange systems improve accessibility for economically disadvantaged students [25]. Dynamic user interfaces significantly improve user engagement and navigation efficiency [26]. Cloud-ready backend architectures provide better scalability for increasing user demands [27].

Structured database relationships ensure consistency between users, books, and requests [28]. The integration of frontend, backend, and database technologies creates efficient end-to-end systems for digital marketplaces [29]. Therefore, existing literature strongly supports the development of a centralized used-book marketplace system that combines donation, lending, request tracking, and secure communication in a scalable and user-friendly environment [30].

III. PROPOSED SYSTEM

The proposed ReBook – Marketplace for Used Books system is a web-based platform developed to provide a secure, organized, and user-friendly environment for exchanging used books through donation and lending. The system addresses the limitations of traditional book exchange methods by introducing a centralized platform where users can register, upload books, search for available books, and send requests for borrowing or donation. The platform is designed using modern full-stack technologies including React for frontend development, Spring Boot and Spring JPA for backend processing, and MySQL for database management. The frontend interface provides responsive navigation and dynamic rendering, enabling users to interact with the system efficiently. The backend manages user authentication, business logic, request processing, and communication between frontend and database layers. The system stores user information, book details, request statuses, and message records securely in the database. Advanced search functionality allows users to locate books using title, author name, ISBN number, publisher details, or keywords. The proposed system also includes integrated messaging functionality for direct communication between users regarding book availability and request handling.



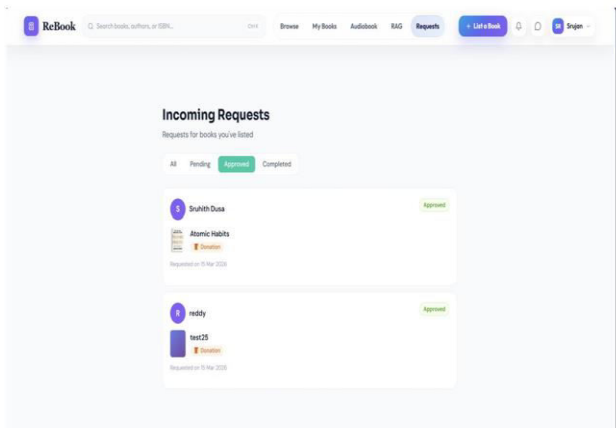
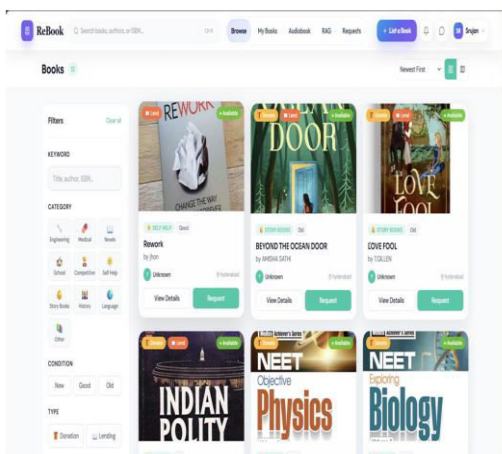
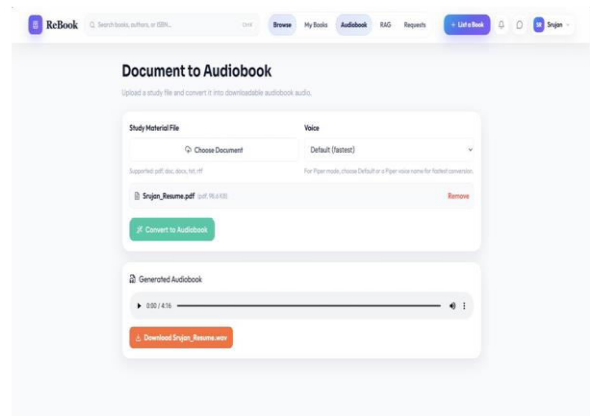
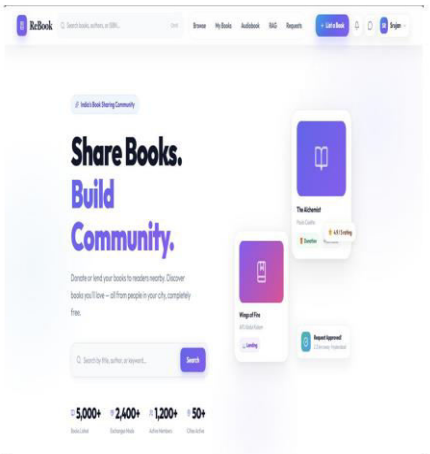
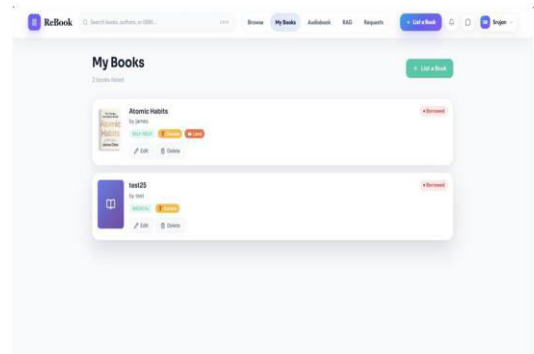
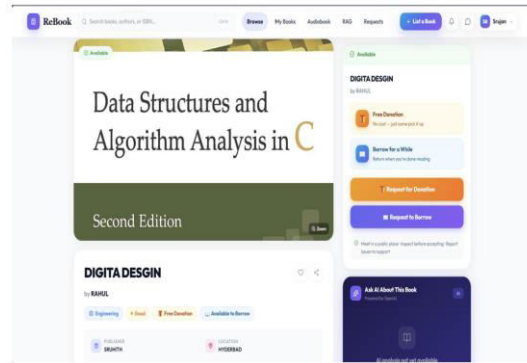
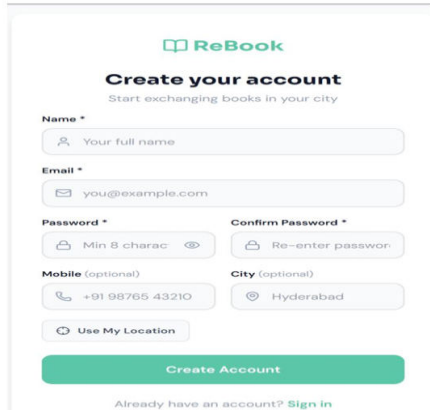
The system introduces structured workflows for donation and lending management to ensure transparency and accountability. Users can send requests for books, while book owners can approve or reject those requests based on availability and preferences. For lending transactions, the system tracks the number of lending weeks and supports return confirmation from both borrower and owner. The application also prevents duplicate requests and unauthorized access through backend validation mechanisms. Role-based functionalities improve system organization by separating user operations and administrative activities. The centralized architecture improves accessibility by allowing users to access the platform anytime and from any location with internet connectivity. The proposed system promotes sustainable reuse of books, reduces paper waste, and helps students minimize educational expenses by accessing affordable or free books. The modular design improves maintainability and scalability, allowing future enhancements such as mobile integration, recommendation systems, and AI-based search optimization. Overall, the proposed ReBook system provides a reliable, scalable, and transparent solution for used book exchange and resource sharing.

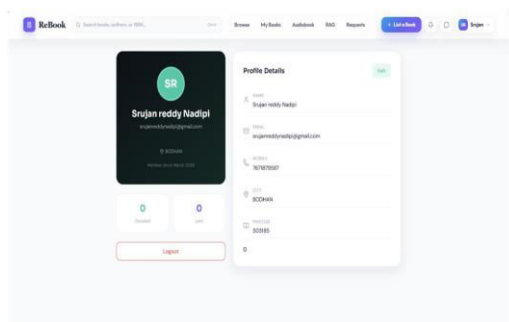
IV. SYSTEM DESIGN

The system design of the ReBook – Marketplace for Used Books application follows a structured three-tier architecture consisting of frontend, backend, and

cloud deployment, and mobile application integration, making the system scalable and adaptable for large-scale usage.

V. RESULTS





VI. CONCLUSION

The ReBook – Marketplace for Used Books system successfully provides a structured, secure, and user-friendly platform for exchanging used books through donation and lending functionalities. The system addresses the major limitations of traditional book exchange methods such as lack of organization, limited accessibility, absence of request tracking, and poor communication between users. By integrating modern full-stack technologies including React, Spring Boot, Spring JPA, and MySQL, the application ensures efficient frontend interaction, secure backend processing, and reliable database management. The platform enables users to register, upload books, search books using advanced filters, send donation or lending requests, and communicate directly through an integrated messaging system. The implementation of request approval workflows, lending duration tracking, and return confirmation mechanisms improves transparency, accountability, and reliability among users. The system also enhances accessibility by allowing users to access educational resources anytime and from any location through an internet-enabled environment. In addition to providing a cost-effective solution for students and readers, the application promotes environmental sustainability by encouraging reuse of books and reducing paper waste. The modular and scalable architecture supports future enhancements such as mobile application support, AI-based recommendation systems, and cloud integration. System testing

confirmed that all modules including registration, authentication, search functionality, request handling, and messaging operate accurately according to functional requirements. The ReBook application demonstrates the practical implementation of software engineering principles, database management, and modern web development technologies in solving real-world educational resource-sharing challenges. Overall, the proposed system provides an efficient, transparent, and sustainable digital marketplace for used book exchange and contributes significantly toward affordable education and resource optimization.

References

1. Facebook Inc. (2023). *React – A JavaScript library for building user interfaces*. Facebook Developer Documentation.
2. Johnson, P., & Hoeller, M. (2023). *Spring Boot framework for rapid application development*. Springer Publications.
3. King, G. (2015). *Java persistence with Hibernate*. Manning Publications.
4. Oracle Corporation. (2023). *MySQL 8.0 reference manual*. Oracle Documentation.
5. Fielding, R. T. (2000). *Architectural styles and the design of network-based software architectures*. University of California.
6. Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson Education.
7. Pressman, R. S. (2014). *Software engineering: A practitioner's approach* (8th ed.). McGraw-Hill.
8. Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns*:

- Elements of reusable object-oriented software*. Addison-Wesley.
9. Deitel, P., & Deitel, H. (2017). *Java how to program* (11th ed.). Pearson.
 10. Freeman, E., Robson, E., Sierra, K., & Bates, B. (2021). *Head first design patterns*. O'Reilly Media.
 11. Fowler, M. (2018). *Refactoring: Improving the design of existing code* (2nd ed.). Addison-Wesley.
 12. Evans, E. (2004). *Domain-driven design: Tackling complexity in the heart of software*. Addison-Wesley.
 13. Richardson, C. (2018). *Microservices patterns*. Manning Publications.
 14. Martin, R. C. (2017). *Clean architecture: A craftsman's guide to software structure and design*. Prentice Hall.
 15. Shklar, L., & Rosen, R. (2009). *Web application architecture: Principles, protocols and practices*. Wiley.
 16. Welling, L., & Thomson, L. (2017). *PHP and MySQL web development* (5th ed.). Addison-Wesley.
 17. Elmasri, R., & Navathe, S. (2016). *Fundamentals of database systems* (7th ed.). Pearson.
 18. Silberschatz, A., Korth, H., & Sudarshan, S. (2019). *Database system concepts* (7th ed.). McGraw-Hill.
 19. Gamma, E. (2015). *Object-oriented analysis and design*. Addison-Wesley.
 20. Date, C. J. (2019). *An introduction to database systems* (8th ed.). Pearson.
 21. Lafore, R. (2017). *Data structures and algorithms in Java*. Sams Publishing.
 22. Goodrich, M., Tamassia, R., & Goldwasser, M. (2014). *Data structures and algorithms in Java*. Wiley.
 23. Tanenbaum, A. S., & Bos, H. (2015). *Modern operating systems* (4th ed.). Pearson.
 24. Stallings, W. (2018). *Network security essentials* (6th ed.). Pearson.
 25. Kurose, J. F., & Ross, K. W. (2021). *Computer networking: A top-down approach* (8th ed.). Pearson.
 26. Hoffer, J. A., George, J. F., & Valacich, J. S. (2016). *Modern systems analysis and design*. Pearson.
 27. Bruegge, B., & Dutoit, A. H. (2018). *Object-oriented software engineering*. Pearson.
 28. Balagurusamy, E. (2020). *Programming with Java*. McGraw-Hill Education.
 29. Duckett, J. (2014). *JavaScript and JQuery: Interactive front-end web development*. Wiley.
 30. Sharma, N., & Patel, R. (2022). "Web-based resource sharing systems for sustainable education." *International Journal of Advanced Computer Science and Applications*, 13(5), 112–120.